



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

| APPLICATION NO. | FILING DATE | FIRST NAMED INVENTOR | ATTORNEY DOCKET NO. | CONFIRMATION NO. |
|-----------------|-------------|----------------------|---------------------|------------------|
| 10/763,585 | 01/23/2004 | Luis Felipe Cabrera | 13768.474 | 7734 |

47973 7590 09/20/2006

WORKMAN NYDEGGER/MICROSOFT
1000 EAGLE GATE TOWER
60 EAST SOUTH TEMPLE
SALT LAKE CITY, UT 84111

EXAMINER

MEHRMANESH, ELMIRA

ART UNIT

PAPER NUMBER

2113

DATE MAILED: 09/20/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

| | | | |
|------------------------------|--------------------------------------|---------------------------------------|--|
| Office Action Summary | Application No. 10/763,585 | Applicant(s) CABRERA ET AL. | |
| | Examiner Elmira Mehrmanesh | Art Unit 2113 | |

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) ☒ Responsive to communication(s) filed on 23 January 2004.
- 2a) ☐ This action is **FINAL**. 2b) ☒ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) ☒ Claim(s) 1-40 is/are pending in the application.
- 4a) Of the above claim(s) _____ is/are withdrawn from consideration.
- 5) ☐ Claim(s) _____ is/are allowed.
- 6) ☒ Claim(s) 1-40 is/are rejected.
- 7) ☐ Claim(s) _____ is/are objected to.
- 8) ☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☒ The drawing(s) filed on 23 January 2004 is/are: a) ☒ accepted or b) ☐ objected to by the Examiner.
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some * c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
 2. ☐ Certified copies of the priority documents have been received in Application No. _____.
 3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).
- * See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

- | | |
|---|---|
| 1) <input checked="" type="checkbox"/> Notice of References Cited (PTO-892) | 4) <input type="checkbox"/> Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ |
| 2) <input type="checkbox"/> Notice of Draftsperson's Patent Drawing Review (PTO-948) | 5) <input type="checkbox"/> Notice of Informal Patent Application |
| 3) <input checked="" type="checkbox"/> Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date _____ | 6) <input type="checkbox"/> Other: _____ |

DETAILED ACTION

The application of Cabrera et al., for a "Selectable data field consistency checking" filed January 23, 2004, has been examined.

Claims 1-40 are presented for examination.

Information disclosed and listed on PTO 1449 has been considered.

Claims 1, 2, 12-18, and 22-31 are rejected under 35 USC § 102.

Claims 3-11, 19-21, and 32-40 are rejected under 35 USC § 103.

Claim Rejections - 35 USC § 102

The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

A person shall be entitled to a patent unless –

(b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

Claims 1, 2, 12-18, and 22-31 are rejected under 35 U.S.C. 102(b) as being anticipated by Koizumi et al. (U.S. Patent No. 4,789,986).

As per claim 1, Koizumi discloses in a computing system that includes one or more processors (Fig. 1), and a system memory (Fig. 2, element 405), wherein the computing system is capable of using the one or more processors to instantiate in the system memory an instance of an application program and a system consistency management module (Fig. 2, element 403), a method for the system consistency

management module to selectively performing consistency checking, the method comprising the following

an act of identifying a plurality of data fields of state information corresponding to the instance that are to be subject to consistency checking, the plurality of data fields (Figs. 4, 5, 6) representing less than all of the state information corresponding to the instance (Fig. 10a, element 1011)

an act of identifying an event that will prompt the consistency checking (Fig. 10a, element 1014)

an act of determining that the event has occurred (Fig. 10b, element 1027)

and in response to determining that the event has occurred, an act of performing the consistency checking on the plurality of data fields (Fig. 10b, element 1026).

As per claim 2, Koizumi discloses an act of determining that the plurality of data fields contains at least one inconsistency identified during the consistency checking performance (Fig. 10b, element 1027)

and in response to the determination that the plurality of data fields contains at least one inconsistency comprises an act of setting the state information of the instance to reflect that the instance is in recovery mode (Fig. 10b, element 1025).

As per claim 12, Koizumi discloses the event is the loading of the state information for the instance from persistent media to system memory (Fig. 10a).

As per claim 13, Koizumi discloses the event is the saving of the state information for the instance to the persistent media (Fig. 7).

As per claim 14, Koizumi discloses the event is the backing up of the state information (Fig. 10b).

As per claim 15, Koizumi discloses the event is the occurrence of a specific time (Fig. 10b).

As per claim 16, Koizumi discloses the event is the passage of a specific amount of time since consistency checking was last performed (Fig. 10b).

As per claim 17, Koizumi discloses computer program product for use in a computing system that includes one or more processors (Fig. 1), and a system memory (Fig. 2, element 405), wherein the computing system is capable of using the one or more processors to instantiate in the system memory an instance of an application program and a system consistency management module (Fig. 2, element 403), the computer program product for performing a method for the system consistency management module to perform selective consistency checking (Fig. 10a, element 1011), the computer program product comprising one or more computer-readable media having thereon computer-executable instructions that, when executed by the one or more processors, cause the computing system to perform the following:

an act of identifying a plurality of data fields of state information corresponding to the instance that are to be subject to consistency checking, the plurality of data fields (Figs. 4, 5, 6) representing less than all of the state information corresponding to the instance (Fig. 10a, element 1011)

an act of identifying an event that will prompt the consistency checking (Fig. 10a, element 1014)

an act of determining that the event has occurred (Fig. 10b, element 1027)

and in response to determining that the event has occurred, an act of performing the consistency checking on the plurality of data fields (Fig. 10b, element 1026).

As per claim 18, Koizumi discloses the one or more computer-readable media further have thereon computer-executable instructions that, when executed by the one or more processors, cause the computing system to further performing the following:

an act of determining that the plurality of data fields contains at least one inconsistency identified during the consistency checking performance (Fig. 10b, element 1027)

and in response to the determination that the plurality of data fields contains at least one inconsistency comprises an act of setting the state information of the instance to reflect that the instance is in recovery mode (Fig. 10b, element 1025).

As per claim 22, Koizumi discloses the one or more computer-readable media comprise physical memory media (Fig. 2).

As per claim 23, Koizumi discloses the physical memory media comprises persistent media (Fig. 7).

As per claim 24, Koizumi discloses the physical memory media comprises system memory (Fig. 1).

As per claim 25, Koizumi discloses in a computing system that includes one or more processors (Fig. 1), and a system memory (Fig. 2, element 405), wherein the computing system is capable of using the one or more processors to instantiate in the system memory an instance of an application program and a system consistency management module (Fig. 2, element 403), a method for the system consistency management module to performing consistency checking as specified by the instance of the application program, the method comprising the following

a step for determining that it is appropriate to perform consistency checking on a plurality of data fields, the plurality of data fields (Figs. 4, 5, 6) representing less than all of the state information corresponding to the instance (Fig. 10a, element 1011)

in response to determining that the event has occurred, an act of performing the consistency checking on the plurality of data fields (Fig. 10b, element 1026).

As per claim 26, Koizumi discloses the step for determining that it is appropriate to perform consistency checking on a plurality of data fields comprises the following:

- an act of identifying the plurality of data fields (Figs. 4, 5, 6) and (Fig. 10a, 10b)
- an act of identifying an event that will prompt the consistency checking (Fig. 10a, element 1014)
- an act of determining that the event has occurred (Fig. 10b, element 1027).

As per claim 27, Koizumi discloses the event is the loading of the state information for the instance from persistent media to system memory (Fig. 10a).

As per claim 28, Koizumi discloses the event is the saving of the state information for the instance to the persistent media (Fig. 7).

As per claim 29, Koizumi discloses the event is the backing up of the state information (Fig. 10b).

As per claim 30, Koizumi discloses the event is the occurrence of a specific time (Fig. 10b).

As per claim 31, Koizumi discloses the event is the passage of a specific amount of time since consistency checking was last performed (Fig. 10b).

Claim Rejections - 35 USC § 103

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

The factual inquiries set forth in *Graham v. John Deere Co.*, 383 U.S. 1, 148 USPQ 459 (1966), that are applied for establishing a background for determining obviousness under 35 U.S.C. 103(a) are summarized as follows:

1. Determining the scope and contents of the prior art.
2. Ascertaining the differences between the prior art and the claims at issue.
3. Resolving the level of ordinary skill in the pertinent art.
4. Considering objective evidence present in the application indicating obviousness or nonobviousness.

Claims 4-11, 20-21, and 32-40 are rejected under 35 U.S.C. 103(a) as being unpatentable over Koizumi et al. (U.S. Patent No. 4,789,986) in view of Choquier et al. (U.S. Patent No. 5,951,694).

As per claim 4, Koizumi discloses the act of identifying a plurality of data fields of state information corresponding to the instance that are to be subject to consistency checking (Fig. 10a, 10b) comprises the following:

Koizumi et al. fails to explicitly disclose an API.

Choquier teaches:

an act of receiving a function call (Fig. 5A, element 502a) using an Application Program Interface from the instance, the function call identifying at least implicitly the plurality of data fields (col. 12, lines 49-67).

It would have been obvious to one of ordinary skill in the art at the time the invention to use the method for checking consistency of distributed data of Koizumi et al. in combination with the service session redirection method of Choquier et al. to detect data inconsistencies.

One of ordinary skill in the art at the time the invention would have been motivated to make the combination because Koizumi et al. discloses a method for checking consistency of distributed data by detecting and eliminating inconsistent data in a system (Fig. 2). Choquier et al. discloses redundant data sets to ensure consistency between such locally-stored data sets (col. 10, lines 4-8). Choquier et al. also uses an application programming interface to communicate with server portions of service applications (col. 12, lines 49-67).

As per claim 5, Koizumi et al. fails to explicitly disclose an API.

Choquier teaches:

the act of receiving a function call (Fig. 5A, element 502a) using an Application Program Interface comprises the following:

an act of receiving the function call via one or more intermediary modules (Fig. 5A).

As per claim 6, Koizumi et al. fails to explicitly disclose an API.

Choquier teaches:

the act of receiving the function call (Fig. 5A, element 502a) via one or more intermediary modules comprises the following: an act of receiving the function call after some restructuring the function call to conform with the Application Program Interface (col. 12, lines 49-67).

As per claim 7, Koizumi discloses the act of identifying an event that will prompt the consistency checking (Fig. 10a, 10b) comprises the following:

Koizumi et al. fails to explicitly disclose an API.

Choquier teaches:

an act of receiving a function call (Fig. 5A, element 502a) using the Application Program Interface from the instance (col. 12, lines 49-67) the function call identifying at least implicitly the event (Fig. 5A).

As per claim 8, Koizumi et al. fails to explicitly disclose an API.

Choquier teaches:

the act of receiving a function call (Fig. 5A, element 502a) using an Application Program Interface comprises the following:

an act of receiving the function call via one or more intermediary modules (Fig. 5A).

As per claim 9, Koizumi et al. fails to explicitly disclose an API.

Choquier teaches:

the act of receiving the function call (Fig. 5A, element 502a) via one or more intermediary modules comprises the following: an act of receiving the function call after some restructuring the function call to conform with the Application Program Interface (col. 12, lines 49-67).

As per claim 10, Koizumi et al. fails to explicitly disclose an API.

Choquier teaches:

the function call identifying the plurality of data fields and the function call identifying the event is the same function call (col. 12, lines 49-67).

As per claim 11, Koizumi discloses identifying an event that will prompt the consistency checking (Fig. 10a, 10b) comprises the following:

Koizumi et al. fails to explicitly disclose an API.

Choquier teaches:

an act of receiving a function call (Fig. 5A, element 502a) using the Application Program Interface from the instance (col. 12, lines 49-67) the function call identifying at

least implicitly the event (Fig. 5A).

As per claim 20, Koizumi discloses computer program product in accordance with claim 17, wherein the computer-executable instructions for performing the act of identifying a plurality of data fields of state information corresponding to the instance that are to be subject to consistency checking (Fig. 10a, 10b) comprise computer-executable instructions for performing the following:

Koizumi et al. fails to explicitly disclose an API.

Choquier teaches:

an act of receiving a function call (Fig. 5A, element 502a) using an Application Program Interface from the instance, the function call identifying at least implicitly the plurality of data fields (col. 12, lines 49-67).

As per claim 21, Koizumi discloses computer-executable instructions for performing the act of identifying an event that will prompt the consistency checking (Fig. 10a, 10b) comprise computer-executable instructions for performing the following:

Koizumi et al. fails to explicitly disclose an API.

Choquier teaches:

an act of receiving a function call (Fig. 5A, element 502a) using the Application Program Interface from the instance (col. 12, lines 49-67) the function call identifying at least implicitly the event (Fig. 5A).

As per claim 32, Koizumi discloses in a computing system that includes one or more processors (Fig. 1), and a system memory (Fig. 2, element 405), wherein the computing system is capable of using the one or more processors to instantiate in the system memory an instance of an application program and a system consistency management module (Fig. 2, element 403), a method for the instance to control whether or not the system consistency management module performs consistency checking, the method comprising the following:

the instance has found an inconsistency in a plurality of data fields (Figs. 4, 5, 6) representing less than all of the state information corresponding to the instance (Fig. 10a, element 1011)

and in response, an act of setting the state information of the instance to reflect that the instance is in recovery mode (Fig 10a, 10b)

Koizumi et al. fails to explicitly disclose an API.

Choquier teaches:

an act of receiving a function call (Fig. 5A, element 502a) using an Application Program Interface from the instance (col. 12, lines 49-67).

As per claim 33, Koizumi discloses the system consistency management module perform consistency checking (Fig. 10a, 10b) on the plurality of data fields (Figs. 4, 5, 6).

Koizumi et al. fails to explicitly disclose an API.

Choquier teaches:

an act of receiving a function call (Fig. 5A, element 502a) using an Application Program Interface from the instance (col. 12, lines 49-67).

As per claim 34, Koizumi discloses an act of identifying an event that will prompt the consistency checking (Fig. 10a, element 1014)

an act of determining that the event has occurred (Fig. 10b, element 1027)

and in response to determining that the event has occurred, an act of performing the consistency checking on the plurality of data fields (Fig. 10b, element 1026).

As per claim 35, Koizumi discloses computer program product of use in a computing system that includes one or more processors (Fig. 1), and a system memory (Fig. 2, element 405), wherein the computing system is capable of using the one or more processors to instantiate in the system memory an instance of an application program and a system consistency management module (Fig. 2, element 403), the computer program product comprising one or more computer-readable media having thereon computer-executable instructions for performing a method for instance to control whether or not the system consistency management module performs consistency checking (Fig. 10a, 10b), the one or more computer-readable media having thereon computer-executable instructions that, when executed by the one or more processors, causes the system consistency management module to perform the following:

the instance has found an inconsistency in a plurality of data field representing less than all of the state information corresponding to the instance (Fig. 10a, element 1014) and in response, an act of setting the state information of the instance to reflect that the instance is in recovery mode (Fig. 10a, 10b)

Koizumi et al. fails to explicitly disclose an API.

Choquier teaches:

an act of receiving a function call (Fig. 5A, element 502a) using an Application Program Interface from the instance (col. 12, lines 49-67).

As per claim 36, Koizumi discloses the system consistency management module perform consistency checking (Fig. 10a, 10b) on the plurality of data fields (Figs. 4, 5, 6).

Koizumi et al. fails to explicitly disclose an API.

Choquier teaches:

the function call is a first function call, and the one or more computer-readable media further have thereon computer-executable instructions that, when executed by the one or more processors, causes the computing system to further perform the following: an act of receiving a function call (Fig. 5A, element 502a) using an Application Program Interface from the instance (col. 12, lines 49-67).

As per claim 37, Koizumi discloses computer program product in accordance with claim 35, wherein the one or more computer-readable media further have thereon

computer-executable instructions that, when executed by the one or more processors, causes the computing system to further perform the following:

- an act of identifying an event that will prompt the consistency checking (Fig. 10a, element 1014)

- an act of determining that the event has occurred (Fig. 10b, element 1027)

- and in response to determining that the event has occurred, an act of performing the consistency checking on the plurality of data fields (Fig. 10b, element 1026).

As per claim 38, Koizumi discloses the one or more computer-readable media comprise physical memory media (Fig. 2).

As per claim 39, Koizumi discloses the physical memory media comprises persistent media (Fig. 7).

As per claim 40, Koizumi discloses the physical memory media comprises system memory (Fig 1).

Claims 3, and 19 are rejected under 35 U.S.C. 103(a) as being unpatentable over Koizumi et al. (U.S. Patent No. 4,789,986) in view of Archibald et al. (U.S. Patent No. 6,928,578).

As per claim 3, Koizumi discloses the act of determining that the plurality of fields (Figs. 4, 5, 6) contains at least one inconsistency comprises the following:

Koizumi et al. fails to explicitly disclose CRC.

Archibald teaches:

an act of generating one or more cyclic redundancy checking values corresponding to the plurality of data fields (col. 6, lines 59-61)

an act of comparing the one or more generated cyclic redundancy checking values to one or more stored cyclic redundancy checking values corresponding to the plurality of data field (Fig. 3, element 306)

and an act of determining that there is at least one cyclic redundancy checking value that does not match (Fig. 3, element 313).

As per claim 19, Koizumi discloses the computer-executable instructions for performing the act of determining that the plurality of fields contains at least one inconsistency comprise computer-executable instructions for performing the following:

Koizumi et al. fails to explicitly disclose CRC.

Archibald teaches:

an act of generating one or more cyclic redundancy checking values corresponding to the plurality of data fields (col. 6, lines 59-61)

an act of comparing the one or more generated cyclic redundancy checking values to one or more stored cyclic redundancy checking values corresponding to the plurality of data field (Fig. 3, element 306)

and an act of determining that there is at least one cyclic redundancy checking value that does not match (Fig. 3, element 313).

It would have been obvious to one of ordinary skill in the art at the time the invention to use the method for checking consistency of distributed data of Koizumi et al. in combination with selectable or programmable data consistency checking methodology system of Archibald et al.'s to detect data inconsistencies.

One of ordinary skill in the art at the time the invention would have been motivated to make the combination because Koizumi et al. discloses a method for checking consistency of distributed data by detecting and eliminating inconsistent data in a system (Fig. 2) with each data set having different segments as shown in figures 4-6. Archibald et al. discloses data with different sectors such as a CRC sector (Fig. 2) and (col. 6, lines 59-61).

Related Prior Art

The following prior art is considered to be pertinent to applicant's invention, but nor relied upon for claim analysis conducted above.

Kedem (U.S. Patent No. 6195761), "Method and apparatus for identifying and repairing mismatched data".

Ostrup et al. (U.S. PGPUB No. 20030217133), "Operator-defined consistency checking in a network management system".


Kalmanek et al. (U.S. Patent No. 5524116), "Packet framer".

Conclusion

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Elmira Mehrmanesh whose telephone number is (571) 272-5531. The examiner can normally be reached on 8-5 M-F.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Robert W. Beausoliel can be reached on (571) 272-3645. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).


ROBERT W. BEAUSOLIEL
Supervisor
Art Unit 2113
Patent and Trademark Office
U.S. Department of Commerce